



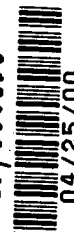
The  
**Patent  
Office**



INVESTOR IN PEOPLE

The Patent Office  
Concept House  
Cardiff Road  
Newport  
South Wales  
NP10 8QQ

jc682 U.S. PTO  
09/558095



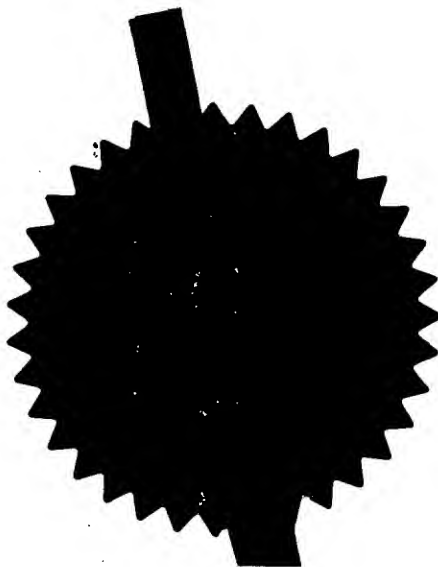
**CERTIFIED COPY OF  
PRIORITY DOCUMENT**

I, the undersigned, being an officer duly authorised in accordance with Section 74(1) and (4) of the Deregulation & Contracting Out Act 1994, to sign and issue certificates on behalf of the Comptroller-General, hereby certify that annexed hereto is a true copy of the documents as originally filed in connection with the patent application identified therein.

In accordance with the Patents (Companies Re-registration) Rules 1982, if a company named in this certificate and any accompanying documents has re-registered under the Companies Act 1980 with the same name as that with which it was registered immediately before re-registration save for the substitution as, or inclusion as, the last part of the name of the words "public limited company" or their equivalents in Welsh, references to the name of the company in this certificate and any accompanying documents shall be treated as references to the name with which it is so re-registered.

In accordance with the rules, the words "public limited company" may be replaced by p.l.c., plc, P.L.C. or PLC.

Re-registration under the Companies Act does not constitute a new legal entity but merely subjects the company to certain additional company law rules.



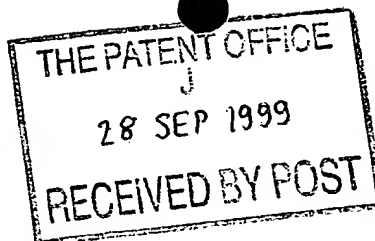
Signed

*W. Evans*

Dated

19 JAN 2000

**This Page Blank (uspto)**



The  
Patent  
Office

1/77

Patents Act 1977  
Rule 16

28SEP99 E479737-1 D00611  
F01/7700 0.00 - 9922802.5

# Request for grant of a patent

The Patent Office

Concept House  
Cardiff Road  
Newport  
South Wales NP9 1RH

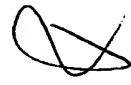
1.	Your reference	UK999104		
2.	Patent application number (The Patent Office will fill in this part)	9922802.5		28 SEP 1999
3.	Full name, address and postcode of the or of each applicant ( <i>underline all surnames</i> )	INTERNATIONAL BUSINESS MACHINES CORPORATION Armonk New York 10504 United States of America		
	Patents ADP number ( <i>if you know it</i> )	519637001 <i>IS</i>		
	If the applicant is a corporate body, give the country/state of its incorporation	State of New York United States of America		
4.	Title of the invention	PUBLISH/SUBSCRIBE DATA PROCESSING WITH PUBLICATION POINTS FOR CUSTOMISED MESSAGE PROCESSING		
5.	Name of your agent ( <i>if you have one</i> )	Kevin J Fournier		
	"Address for Service" in the United Kingdom to which all correspondence should be sent ( <i>including the postcode</i> )	IBM United Kingdom Limited Intellectual Property Department Hursley Park Winchester Hampshire SO21 2JN		
	Patents ADP number ( <i>if you know it</i> )	7257348001 <i>IS</i>		
6.	If you are declaring priority from one or more earlier patent applications, give the country and the date of filing of the or of each of these earlier applications and ( <i>if you know it</i> ) the or each application number	Country	Priority App No ( <i>if you know it</i> )	Date of filing ( <i>day/month/year</i> )
7.	If this application is divided or otherwise derived from an earlier UK application, give the number and the filing date of the earlier application	No of earlier application		Date of filing ( <i>day/month/year</i> )

8. Is a statement of inventorship and of right to grant of a patent required in support of this request? (Answer 'Yes' if:  
a) any applicant named in part 3 is not an inventor, or  
b) there is an inventor who is not named as an applicant, or  
c) any named applicant is a corporate body.)
- Yes

9. Enter the number of sheets for any of the following items you are filing with this form. Do not count copies of the same document

Continuation sheets of this form

Description	9
Claim(s)	2
Abstract	1
Drawing(s)	3



10. If you are also filing any of the following, state how many against each item.

Priority documents

Translations of priority documents

Statement of inventorship and right to grant of a patent (Patents Form 7/77) 3

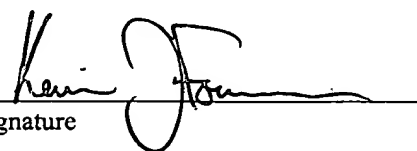
Request for preliminary examination and search (Patents Form 9/77) 1

Request for substantive examination (Patents Form 10/77)

Any other documents (please specify)

Fee sheet (FS1)

11. I/We request the grant of a patent on the basis of this application

Signature 

27 September 1999  
Date

12. Name and daytime telephone number of person to contact in the United Kingdom Kevin J Fournier  
01962 816574

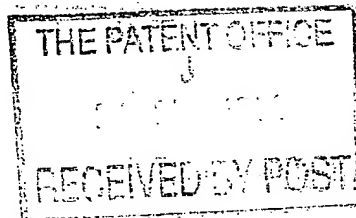
The  
Patent  
Office

7/77

Patents Act 1977

Rule 15

Statement of inventorship and of  
right to grant of a patent



The Patent Office

Concept House  
Cardiff Road  
Newport  
South Wales NP9 1RH

1.	Your reference	UK999104
2.	Patent application number (if you know it)	9922802.5 28 SEP 1999
3.	Full name of the or of each applicant	INTERNATIONAL BUSINESS MACHINES CORPORATION
4.	Title of invention	PUBLISH/SUBSCRIBE DATA PROCESSING WITH PUBLICATION POINTS FOR CUSTOMISED MESSAGE PROCESSING
5.	State how the applicant(s) derived the right from the inventor(s) to be granted a patent	By employment and by agreement
6.	How many, if any, additional Patents Forms 7/77 are attached to this form?	
7.	I/We believe that the person(s) named over the page (and on any extra copies of this form) is/are the inventor(s) of the invention which the above patent application relates to.	
	Signature	27 September 1999 Date
8.	Name and daytime telephone number of person to contact in the United Kingdom	Kevin J Fournier 01962 816574

Enter the full names, addresses and postcodes of the inventors in the boxes and underline the surnames

HOLDSWORTH, Simon A J  
UK Resident  
c/o IBM United Kingdom Limited  
Intellectual Property Law  
Hursley Park  
Winchester  
Hampshire  
SO21 2JN  
UK

Patents ADP number (if known)

7749682001

TODD, Stephen J  
UK Resident  
c/o IBM United Kingdom Limited  
Intellectual Property Law  
Hursley Park  
Winchester  
Hampshire  
SO21 2JN  
UK

Patents ADP number (if known)

7702020001

If there are more than three inventors, please write their names and addresses on the back of another Patents Form 7/77 and attach it to this form

AYRES, Malcolm  
UK Resident  
c/o IBM United Kingdom Limited  
Intellectual Property Law  
Hursley Park  
Winchester  
Hampshire  
SO21 2JN  
UK

Patents ADP number (if known)

7749658001

**REMINDER**

Have you signed the form?

*IS*

Enter the full names, addresses and postcodes of the inventors in the boxes and underline the surnames

HOLLOWAY, Timothy  
UK Resident  
c/o IBM United Kingdom Limited  
Intellectual Property Law  
Hursley Park  
Winchester  
Hampshire  
SO21 2JN  
UK

Patents ADP number (if known)

7749674001

SCHMIDT, Marc-Thomas  
UK Resident  
c/o IBM United Kingdom Limited  
Intellectual Property Law  
Hursley Park  
Winchester  
Hampshire  
SO21 2JN  
UK

Patents ADP number (if known)

7686447001

If there are more than three inventors, please write their names and addresses on the back of another Patents Form 7/77 and attach it to this form

TAYLOR, Michael G  
UK Resident  
c/o IBM United Kingdom Limited  
Intellectual Property Law  
Hursley Park  
Winchester  
Hampshire  
SO21 2JN  
UK

Patents ADP number (if known)

7749666001

# REMINDER

Have you signed the form?

PUBLISH/SUBSCRIBE DATA PROCESSING WITH PUBLICATION POINTS  
FOR CUSTOMISED MESSAGE PROCESSING

5     Field of the Invention

10     The present invention relates to the field of data processing and more specifically to event notification data processing which distributes event messages from suppliers (called, hereinafter, "publishers") of data messages to consumers (called, hereinafter "subscribers") of such messages. While there are many different types of known event notification systems, the subsequent discussion will describe the publish/subscribe event notification system as it is one of the most common.

15     Background of the Invention

20     Publish/subscribe data processing systems (and event notification systems in general) have become very popular in recent years as a way of distributing data messages (events) from publishing computers to subscribing computers. The increasing popularity of the Internet, which has connected a wide variety of computers all over the world, has helped to make such publish/subscribe systems even more popular. Using the Internet, a World Wide Web browser application (the term "application" or "process" refers to a software program, or portion thereof, running on a computer) can be used in conjunction with the publisher or subscriber in order to graphically display messages. Such systems are especially useful where data supplied by a publisher is constantly changing and a large number of subscribers needs to be quickly updated with the latest data. Perhaps the best example of where this is useful is in the distribution of stock market data.

35     In such systems, publisher applications of data messages do not need to know the identity or location of the subscriber applications which will receive the messages. The publishers need only connect to a publish/subscribe distribution agent process, which is included in a group of such processes making up a broker network, and send messages to the distribution agent process, specifying the subject of the message to the distribution agent process. The distribution agent process then distributes the published messages to subscriber applications which have previously indicated to the broker network that they would like to receive data messages on particular subjects. Thus, the subscribers also do not need to know the identity or location of the publishers. The subscribers need only connect to a distribution agent process.



One such publish/subscribe system which is currently in use, and which has been developed by the Transarc Corp. (a wholly owned subsidiary of the assignee of the present patent application, IBM Corp.) is shown in Fig. 1. Publishers 11 and 12 connect to the publish/subscribe broker network 2 and send published messages to broker network 2 which distributes the messages to subscribers 31, 32, 33, 34. Publishers 11 and 12, which are data processing applications which output data messages, connect to broker network 2 using the well known inter-application data connection protocol known as remote procedure call (or RPC) (other well known protocols, such as asynchronous message queuing protocols, can also be used). Each publisher application could be running on a separate machine, alternatively, a single machine could be running a plurality of publisher applications. The broker network 2 is made up of a plurality of distribution agents (21 through 27) which are connected in a hierarchical fashion which will be described below as a "tree structure". These distribution agents, each of which could be running on a separate machine, are data processing applications which distribute data messages through the broker network 2 from publishers to subscribers. Subscriber applications 31, 32, 33 and 34 connect to the broker network 2 via RPC in order to receive published messages.

Publishers 11 and 12 first connect via RPC directly to a root distribution agent 21 which in turn connects via RPC to second level distribution agents 22 and 23 which in turn connect via RPC to third level distribution agents 24, 25, 26 and 27 (also known as "leaf distribution agents" since they are the final distribution agents in the tree structure). Each distribution agent could be running on its own machine, or alternatively, groups of distribution agents could be running on the same machine. The leaf distribution agents connect via RPC to subscriber applications 31 through 34, each of which could be running on its own machine.

In order to allow the broker network 2 to determine which published messages should be sent to which subscribers, publishers provide the root distribution agent 21 with the name of a distribution stream for each published message. A distribution stream (called hereinafter a "stream") is an ordered sequence of messages having a name (e.g., "stock" for a stream of stock market quotes) to distinguish the stream from other streams (this is known as "topic based" publish/subscribe, another well known model is called "content based publish/subscribe which involves matching publishers and subscribers by the content of the messages rather than by the topic). Likewise, subscribers provide the leaf distribution agents 31 through 34 with the name of the streams to which they would like to subscribe. In this way, the broker network 2 keeps track of which subscribers are interested in which streams so that when publishers publish messages to such streams, the messages can be distributed to the

corresponding subscribers. Subscribers are also allowed to provide filter expressions to the broker network in order to limit the messages which will be received on a particular stream (e.g., a subscriber 31 interested in only IBM stock quotes could subscribe to the stream "stock" by making an RPC call to leaf distribution agent 24 and include a filter expression stating that only messages on the "stock" stream relating to IBM stock should be sent to subscriber 31).

The above-described publish/subscribe architecture provides the advantage of central co-ordination of all published messages, since all publishers must connect to the same distribution agent (the root) in order to publish a message to the broker network. For example, total ordering of published messages throughout the broker network is greatly facilitated, since the root can easily assign sequence numbers to each published message on a stream. However, this architecture also has the disadvantage of publisher inflexibility, since each publisher is constrained to publishing from the single root distribution agent, even when it would be much easier for a publisher to connect to a closer distribution agent.

In the Fig. 1, a publisher application 11, running on one computer, is, for example, a supplier of live stock market data quotes. That is, publisher application 11 provides frequent messages stating the present value of share prices. In this example, publisher application 11 is publishing messages on a stream called "stock" which has already been configured in the broker network 2. As is well known, when publisher 11 wishes to publish a stock quote message to stream "stock", publisher 11 makes an RPC call to the root distribution agent 11 which is at the top level of the broker network tree structure. In this example, subscriber application 32, running on another computer, has sent a subscription request via an RPC call to leaf distribution agent 24, which is at the bottom level of the tree structure, indicating that subscriber 32 would like to subscribe to stream "stock".

Thus, whenever publisher 11 publishes a data message to stream "stock" the distribution tree structure of broker network 2 channels the message down through the root distribution agent 21, through any intermediary distribution agents (e.g., 22 in the example of Fig. 1) and through the leaf distribution agent 24 to the subscriber 32. This involves a series of RPC calls being made between each successive circle in the diagram of Fig. 1 connecting publisher 11 and subscriber 32 (i.e., 11 to 21, 21 to 22, 22 to 24 and 24 to 32).

Figure 2 shows a different publish/subscribe architecture where publisher applications can publish messages to the broker network by directly communicating with any one of a plurality of distribution agents

(brokers). For example, publisher application 201 is shown communicating directly with Broker 12. There is no requirement in this architecture that all publisher applications communicate directly with a top (or root) distribution agent. Publisher application 201 can potentially  
5 communicate directly with any of the distribution agents shown in Fig 2, in the described examples below it will be shown communicating directly with Broker 12.

Subscriber applications 202 and 203 would like to receive messages  
10 on the stream/topic that publisher application 201 is publishing on. Thus, subscriber applications 202 and 203 communicate directly with Brokers 1112 and 1221, respectively, to provide subscription data thereto informing the broker hierarchy of their desire to receive such published  
15 messages. Since the publisher application 201 is allowed to communicate directly with any of a plurality of distribution agents, the subscription data entered by the subscriber applications must be propagated throughout the broker network to each Broker shown in Fig. 2. This way, no matter which distribution agent the publisher application 201 happens to  
20 communicate directly with, the published messages will be able to be routed to the subscriber applications 202 and 203.

Publish/subscribe broker systems have commonly been integrated into multi-function message broker systems which are used to inter-connect applications which may be on heterogeneous platforms and may use  
25 different message formats. For example, Saga Software of Reston, Virginia (USA) ([www.sagasoftware.com](http://www.sagasoftware.com)) have such a message broker product called "Sagavista" (a trademark of Saga Software). Further, Tibco Software Inc. of Palo Alto, California (USA) ([www.tibco.com](http://www.tibco.com)) also have such a message  
30 broker called "TIB/Message Broker" (both "TIB" and "TIB/Message Broker" are trademarks of Tibco). In these multi-function message brokers, a set of pluggable data processing nodes is provided, with each node being dedicated to a specific data processing task, such as message format transformation, publish/subscribe message distribution, and a rules engine for deciding (based on a plurality of predefined rules) where an  
35 incoming message should be routed.

In these multi-function message broker products, when a subscriber application registers a subscription request with the broker, the subscriber application sends the subscription request to a  
40 publish/subscribe broker node specifying the topic of the desired subscription. The publish/subscribe broker node (usually in cooperation with a plurality of other such publish/subscribe broker nodes) then ensures that any published messages on that topic are sent to the subscriber application. Different subscribers may wish to receive the  
45 same published messages but in different message formats (or may desire that some other type of processing be carried out on published messages

before such messages are delivered to the subscriber). For example, a subscriber in the United States may want to know IBM's stock price per share in US dollars while another subscriber in the United Kingdom may want to know IBM's stock price in UK (British) pounds.

In order to accommodate such format desires of various subscribers, the message broker would have to modify the topic after having performed a format transformation so that a subscriber can subscribe to this modified topic (rather than the original topic that the publisher published on) in order to receive the format-transformed messages. Alternatively, the publishers would have to publish the same messages in different formats (with each format having its own topic), thus doing away with the need for the broker to do the format transformation. Because the topic needs to correspond to the format in both of these cases, this can cause many problems. For example, it is very useful to carry out access control on a topic basis. That is, when deciding which subscribers can have access to which published messages, it is very useful to be able to use the topics of the messages to make such access control decisions. However, when the topics must be different for essentially the same group of messages because of format changes, such access control decisions become much more complex.

It would be clearly desirable to be able to use the same topic for a variety of different message formats in a message broker, but the present state of the art does not allow for this.

#### Summary of the Invention

According to one aspect, the present invention provides a message broker data processing apparatus including: message broker data processing apparatus comprising: means for receiving published messages on a topic from a plurality of publisher applications; means for processing the received messages; and means for distributing the processed messages to a subscriber application; wherein the means for receiving includes a plurality of publication point data processing nodes, each of which receives published messages on said topic from a publisher application.

According to a second aspect, the present invention provides a data processing method of carrying out the functionality discussed above with respect to the first aspect.

According to a third aspect, the present invention provides a computer readable storage medium having a computer program stored on it which, when executed on a computer, carries out the functionality of data processing method of the second aspect of the invention.

Thus, the present invention provides a message broker having a publish/subscribe capability where a publisher application can publish messages in a manner which is most convenient to that publisher application, and a subscriber application will receive such published messages after the messages have undergone specific data processing, all without the need for the topic names used by the publisher application, broker and subscriber application to be modified. For example, the publisher, broker and subscriber can use the same topic name even though the messages sent under this topic will be of differing formats. The presence of multiple publication points, selectable by a particular publisher application, within the broker provides for this ability.

As one advantage of the invention, access control can thus be easily carried out using the topic name. Further, the publisher application does not need to publish the same messages on a plurality of topics in order to accommodate subscribers who want publications in differing formats, thus decoupling the publisher application from having to deal with the varying desires of subscribers. The publisher need only publish messages in the format most convenient to that publisher.

#### Brief Description of the Drawings

The invention will be better understood by referring to the detailed description of the preferred embodiments which will now be described in conjunction with the following drawing figures:

Figure 1 is a block diagram showing a first architecture of a publish/subscribe data processing system to which the preferred embodiment of the present invention can be advantageously applied;

Figure 2 is a block diagram showing a second architecture of a publish/subscribe data processing system to which the preferred embodiment of the present invention can be advantageously applied; and

Fig. 3 is a block diagram showing an exemplary message broker according to a preferred embodiment of the present invention.

#### Detailed Description of the Preferred Embodiments

In Fig. 3 a message broker 32 receives published messages on a topic called "IBM stock" from a publisher application 31a (which is an application running at a major stock exchange in the United States of America) and distributes such published messages to subscriber application 33 (which is a stock broking agency also located in the United States of America) which has previously registered a subscription to the topic "IBM stock". Message broker 32 also receives published messages on the topic "IBM stock" from another publisher application 31b

(which is an application running at a major stock exchange in the United Kingdom) and distributes such published messages to subscriber application 33 (again, which is a stock broking agency located in the United States of America) which has previously registered a subscription to the topic "IBM stock". In this example, the publisher application, broker and subscriber applications are all running on separate machines (and are thus interconnected via a network which is not shown in Fig. 3). In other embodiments, however, two or more of the applications (e.g., the publisher and the broker) could be running on the same machine. Further, as was explained above, the broker 32 is most likely running on a plurality of machines.

When one of the publisher applications 31a or 31b communicates with the broker 32 in order to publish messages thereto, the publisher application specifies a particular publication point (e.g., 323 or 324) as the point of entry into the message broker 32. A publication point data processing node (or "publication point" for short) is a data processing node which acts as a point of entry for published messages in a messageflow of data processing nodes making up a message broker. That is, each publication point is at the beginning of a specific data processing path through the broker. A publisher application selects a publication point depending on which particular desired path the published messages should take depending on the nature of the published messages and the nature of the processing that will be carried out on that path.

For example, publisher application 31b selects publication point 324 because publisher application 31b is located in the United Kingdom and thus publisher application 31b "knows" that a message transformation will be needed. Specifically, once the publisher application 31b's published messages pass through the publication point 324, they are passed to message transformation data processing node 321 which performs the function of transforming the format of the published messages so that the IBM stock prices, which are originally published in UK pounds by publisher application 31b, are converted to US dollars. The message transformation node 321 accesses local storage 322 in order to determine the current exchange rate of UK pounds to US dollars (this exchange rate is updated at the beginning of every business day). After having their UK pound amounts converted to US dollars, the messages are output from the message transformation node 321 and received at a subscription point processing node 325.

A subscription point data processing node (or "subscription point" for short) is an instance of a publish/subscribe matching engine which performs the function of looking at the topics in previously received subscription requests (received from subscribers) and determining whether

the topic in an incoming message (just received from a publisher application) matches the topic of any of the previously received subscription requests. For any subscriptions that match, the subscription point data processing node distributes the published message to the subscriber application(s) which had entered the subscription requests.

Back to Fig. 3, the subscription point processing node 325 determines (e.g., by accessing local storage 322) that subscriber application 33 has previously entered a subscription on the topic "IBM stock". Thus, subscription point processing node 325 distributes the published messages to subscriber application 33.

On the other hand, publisher application 31a communicates with the broker 32 via another publication point 323, and thus published messages from publisher 31a take another path through the broker bypassing the message transformation data processing node 321. Specifically, the published messages from publisher 31a are sent directly to subscription point data processing node 325. Publisher application 31a chooses to communicate with publication point 323 because publisher application 31a is located in the United States and thus the published messages are already in the US dollars format, and thus there is no need to transform the messages to the US dollars format, which is the format required by the subscriber application 33. Subscription point data processing node 325 then performs a publish/subscribe topic matching operation and determines that subscriber application 33 has previously entered a subscription request to the topic "IBM stock". Thus, subscription point processing node 325 distributes the published messages to subscriber application 33.

Thus, by the use of a plurality (two in Fig. 3) of publication point data processing nodes in a message broker, publisher applications can select amongst the plurality of publication points in order to publish messages which will be received by subscribers in a message format selected by the subscriber without having to use different topics (the topic "IBM stock" is the same for both publication points 323 and 324 and for both publisher applications 31a and 31b). This allows access control to be easily carried out on a topic basis. For example, the broker can perform a security measure on both publisher applications 31a and 31b by simply checking whether the requested topic "IBM stock" of their published messages is a topic which has previously been determined as acceptable for publishers 31a and 31b from a security standpoint.

In a multi-broker environment the subscription point at each of several brokers is connected to the subscription point at other brokers exactly as described for simple publish/subscribe systems. The message

is published to a publication point at an initial receiving broker (IRB). Broker IRB process the message according to the publication point on which it was published. The processing in broker IRB may cause the message (or one or more derivations of the message) to reach the subscription point at broker IRB. Once a message (original or derivative) reaches the subscription point at broker IRB it is made available to subscribers on other brokers using standard interbroker publish/subscribe technology. This mechanism of this inter broker publish/subscribe technology operates independently from the mechanism by which the message reaches the subscription point at broker IRB.

The use of publication points in message flows through the broker is not limited to knowledge of a downstream message format transformation. Such publication points could be used in a wide variety of different contexts.

**This Page Blank (uspto)**



CLAIMS

1. A message broker data processing apparatus comprising:

means for receiving published messages on a topic from a plurality of publisher applications;

means for processing the received messages; and

means for distributing the processed messages to a subscriber application;

wherein the means for receiving includes a plurality of publication point data processing nodes, each of which receives published messages on said topic from a publisher application.

2. The apparatus of claim 1 wherein said apparatus communicates with a subscriber application over the Internet and wherein at least one of the subscriber applications and the publisher application runs in conjunction with a World Wide Web browser.

3. The apparatus of claim 1 wherein the broker performs different processing on published messages according to the publication point on which said published messages are published.

4. The apparatus of claim 3 wherein messages in different formats are published on different publication points, and the processing associated with each publication point is used to render the messages into a standard format available at a subscription point.

5. A message broker data processing method comprising steps of:

receiving published messages on a topic from a plurality of publisher applications;

processing the received messages; and

distributing the processed messages to a subscriber application;

wherein the receiving step is carried out via a plurality of publication point data processing nodes, each of which receives messages on said topic from a publisher application.

6. The method of claim 5 wherein the broker performs different processing on published messages according to the publication point on which said published messages are published.

5 7. The method of claim 6 wherein messages in different formats are published on different publication points, and the processing associated with each publication point is used to render the messages into a standard format available at a subscription point.

10 8. A computer program product stored on a computer readable storage medium for, when run on a computer, instructing the computer to carry out the method steps recited in claim 5.

15

~~This Page Blank (uspto)~~

## ABSTRACT

5

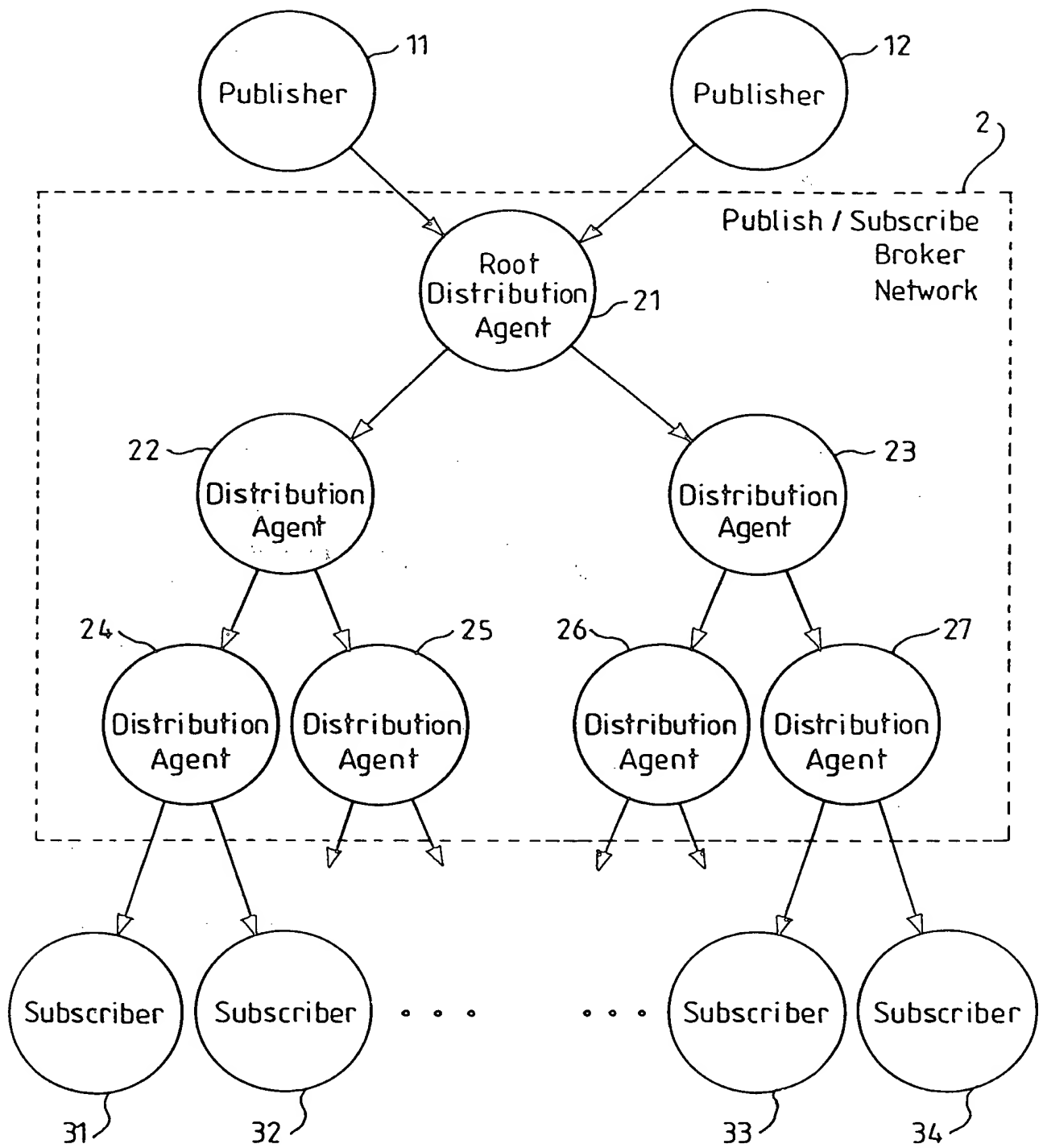
PUBLISH/SUBSCRIBE DATA PROCESSING WITH PUBLICATION POINTS  
FOR CUSTOMISED MESSAGE PROCESSING

10

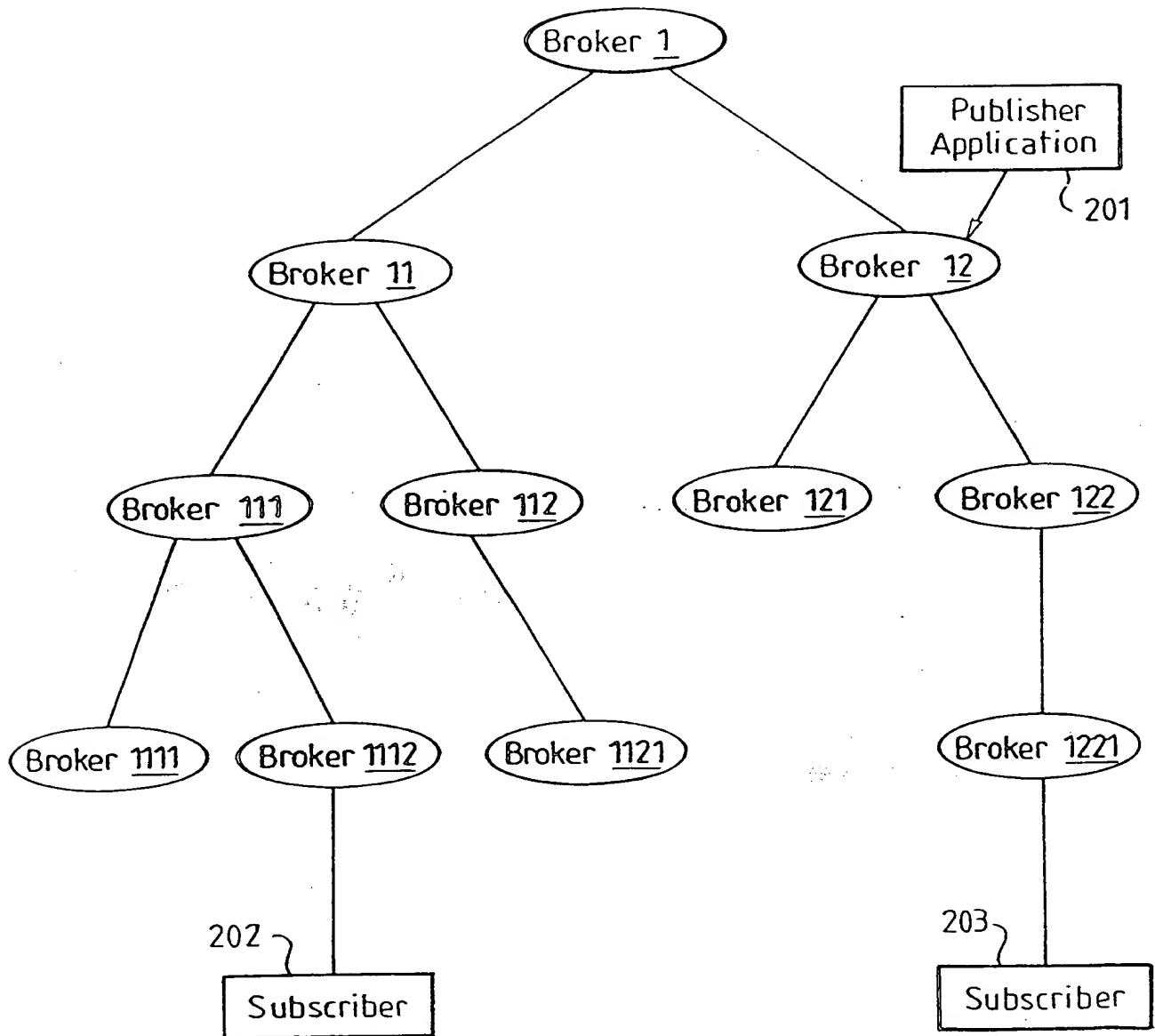
A message broker data processing apparatus comprising: means for receiving published messages on a topic from a plurality of publisher applications; means for processing the received messages; and means for distributing the processed messages to a subscriber application; wherein the means for receiving includes a plurality of publication point data processing nodes, each of which receives published messages on said topic from a publisher application.

15

This Page Blank (uspto)

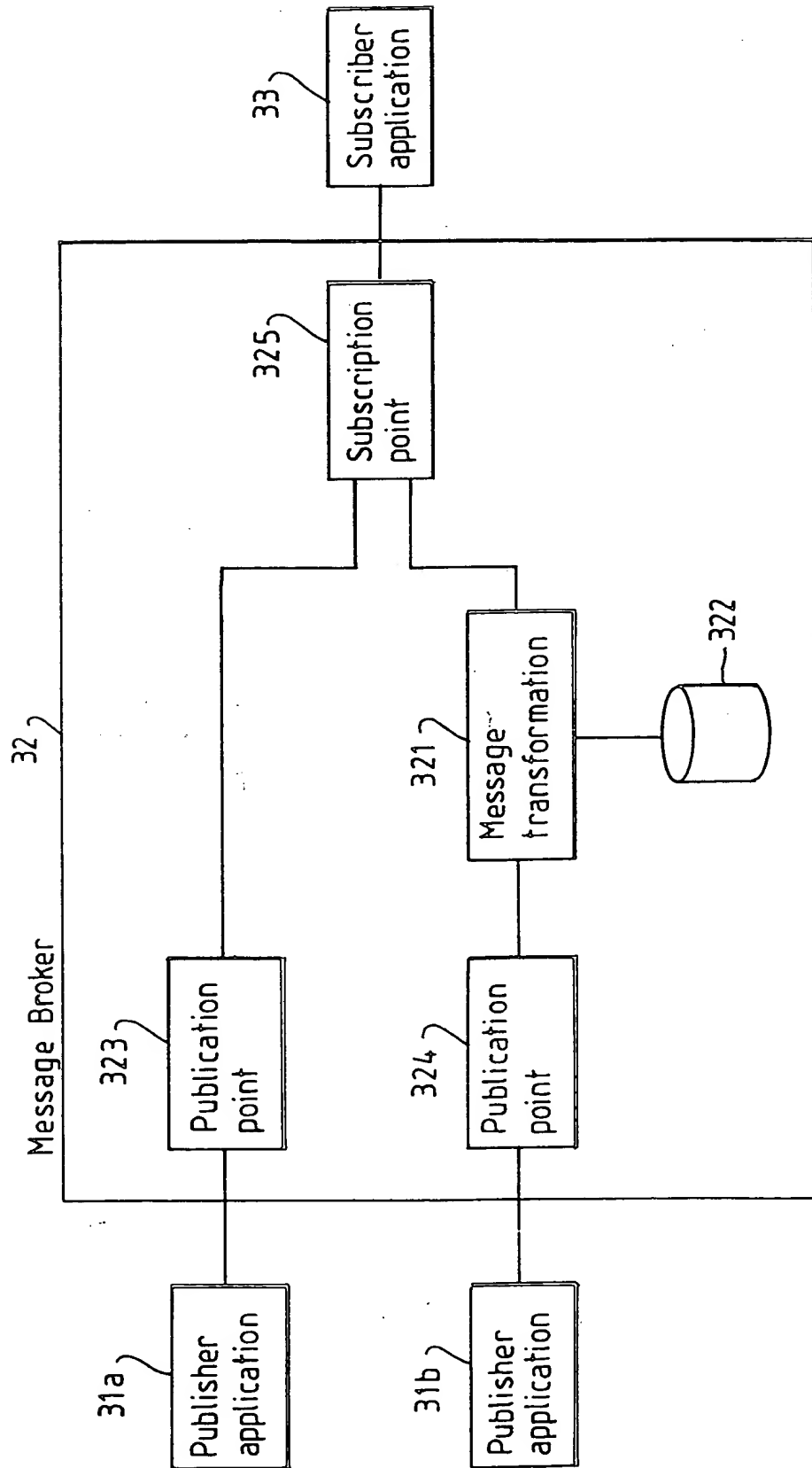
FIG. 1

**This Page Blank (uspto)**

FIG. 2

**This Page Blank (uspto)**



FIG. 3

**This Page Blank (uspto)**